Raport – analiza oprogramowania Cry

Próbka

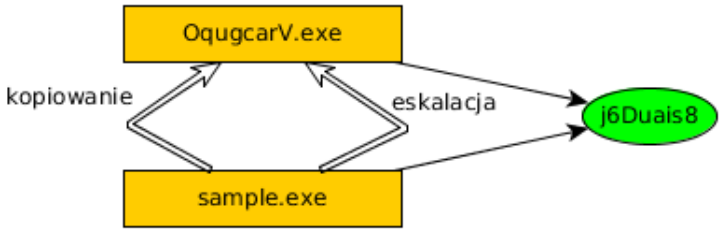
Analizowana próbka to moduł PE32 o następujących skrótach kryptograficznych.

Md5	41ec3fefd7b25b3cde15737a27d16735
Sha256	a6a8b82acdf7d2c873c453f740527d33c758a97c29d4f76cad6843118b7cfaaf
Sha512	138dec7a43d46fc2e4cf279feceb20065bf08a83271e6273eefcd909df495cce1f1ed23a8c6ae123c22dacd8ed999c58621fc9c935d6abb2f9b6defdb425dfba

Próbka została uzyskana w trakcie analizy powłamaniowej w systemie jednego z klientów Orange Polska.

Architektura

Cry to aplikacja typu ransomware. Jej celem jest zaszyfrowanie potencjalnie ważnych danych na komputerze ofiary i żądanie okupu za klucz deszyfrujący. Ofiara otrzymuje komunikat o żądaniu okupu i w efekcie realizuje transakcję na rzecz cyberprzestępców. Na koniec cyberprzestępca przekazuje instrukcje, co zrobić, by odszyfrować dane.

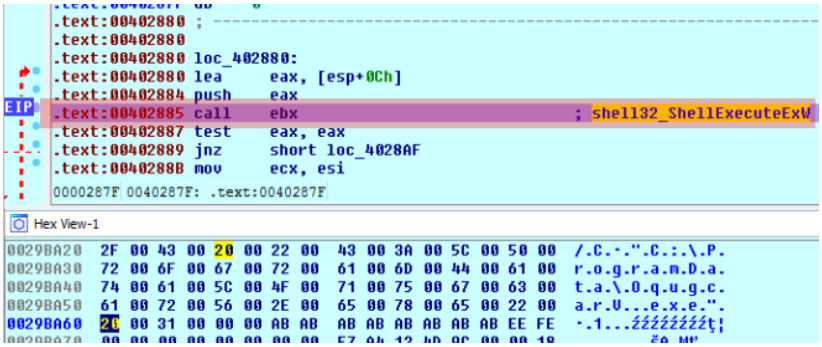


Cry posiada prostą architekturę – jeden moduł wykonywalny, który sam kopiuje siebie w głąb systemu plików, a następnie usiłuje się uruchomić z podwyższonymi uprawnieniami. Podczas działania korzysta on z mutexa synchronizującego działanie oraz tworzy różne klucze rejestru oraz skróty, z których korzysta w trakcie prowadzenia swoich operacji.

Działanie

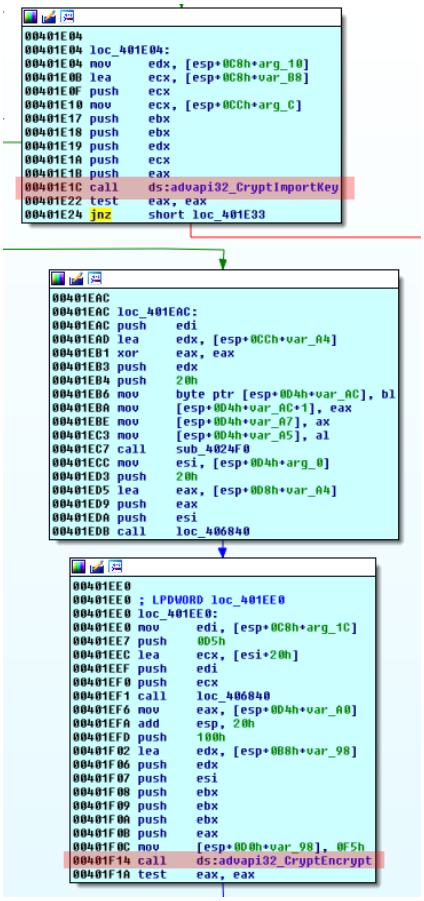
Cry najpierw sprawdza, czy infekcja została już przeprowadzona, poszukując mutexu z wygenerowaną pseudolosową nazwą. Jeśli mutex nie istnieje, malware przystępuje do infekowania komputera.

Infekcja ogranicza się do skopiowania podstawowego modułu do wyznaczonej lokacji w systemie plików oraz utworzenia wskazującego na siebie skrótu w folderze Startup. Nowa nazwa modułu jest generowana przez generator pseudolosowy. Po przeprowadzeniu tych operacji podejmowana jest próba uruchomienia skopiowanego w głąb systemu plików modułu z podwyższonymi uprawnieniami.



Po udanym podwyższeniu uprawnień Cry wysyła raport poprzez kanał C&C, którego konstrukcja została omówiona w dalszej części raportu.

Następnym krokiem jest usunięcie punktów przywracania systemu (shadow volumes) i przystąpienie do szyfrowania potencjalnie cennych plików w systemie ofiary. Klucz wykorzystywany do szyfrowania jest generowany za pomocą systemowych usług kryptograficznych.



Cry posługuje się dwoma parami kluczy RSA. Pierwsza z nich, nazywana dalej kluczami operacyjnymi, jest generowana w procesie infekcji na komputerze ofiary. Jej publiczna część jest wykorzystywana do szyfrowania danych, natomiast część prywatna jest szyfrowana za pomocą publicznej części drugiej pary

kluczy (zwanej dalej kluczami głównymi). Za pomocą szeregi operacji arytmetycznych prywatny klucz operacyjny przekształcany jest do indywidualnego kodu identyfikującego ofiarę.

Tworzona jest również notatka z żądaniem okupu, która jest ustawiana jako tło pulpitu. Aby upenić się, że ofiara zauważy notatkę, wszystkie skróty z pulpitu są przenoszone do katalogu „old_shortcuts", tak, by jej nie zasłaniały.

Identyfikator ofiary jest przechowywany w rejestrze systemowym i można do niego uzyskać dostęp w każdym momencie. W trakcie wpłacania okupu ofiara za pośrednictwem dedykowanej strony w sieci TOR ma go przekazać botmasterowi, który po potwierdzeniu wpłaty okupu rozszyfruje go prywatnym kluczem głównym, co pozwala odzyskać prywatną część klucza operacyjnego w systemie ofiary i odszyfrować dane.

Szyfrowanie

Przed przystąpieniem do szyfrowania i usuwania niezaszyfrowanych plików aplikacja usuwa punkty przywracania systemu Volume Shadow Copies.

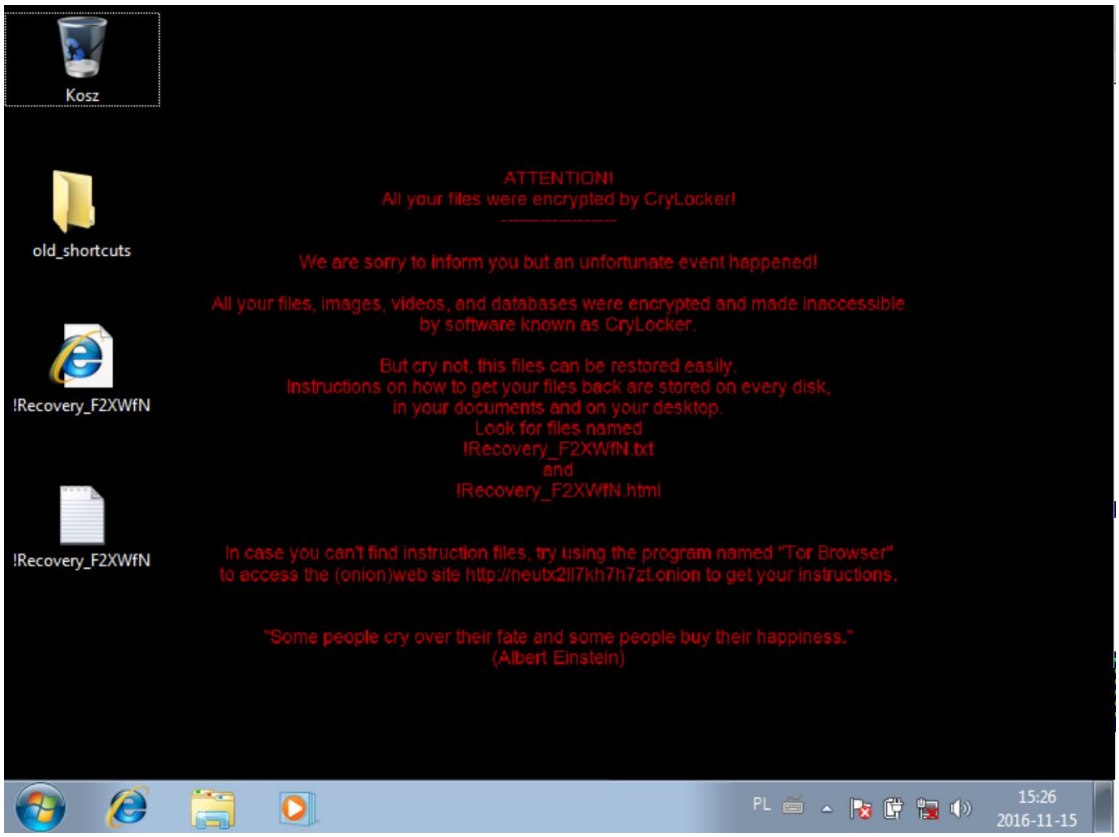
Samo szyfrowanie jest przeprowadzane w dwóch etapach. W pierwszym etapie przeprowadzana jest analiza systemu plików ofiary. Ransomware w rekurencyjnie wykonywanej procedurze przeszukuje strukturę katalogów w poszukiwaniu kandydatów do szyfrowania. Warunkami, które musi spełnić plik jest posiadanie jednego z kilkudziesięciu rozszerzeń, przy czym istnieje również lista nazw plików wykluczonych.

Po utworzeniu listy uruchamiane są wątki, które przeprowadzają szyfrowanie wszystkich plików z listy.

Zawartość pliku najpierw kopiowana jest do katalogu tymczasowego i szyfrowana, a następnie plik docelowy jest zastępowany nową, zaszyfrowaną wersją, podczas, gdy plik w katalogu tymczasowym jest usuwany.

Notka

Po przeprowadzeniu tych operacji, malware informuje ofiarę o przeprowadzonych operacjach i możliwości zapłaty okupu, umieszczając na pulpicie pliki zawierające instrukcje kontaktowania się z botmasterem za pośrednictwem portalu w sieci TOR. Dodatkowo przenosi wszystkie pliki z pulpitu do katalogu old_shortcuts, a tło pulpitu zmienia na notatkę z odniesieniem do tych instrukcji.



Dystrybucja kluczy

Aplikacje typu ransomware posiadają nieco inne kanały C&C w stosunku do złośliwego oprogramowania innych typów (np. RAT, bankers, etc.). Wynika to z faktu, że prowadzenie kampanii ransomware nie wymaga wymiany dużej ilości danych pomiędzy botem a botmasterem, zaś większość złośliwych operacji może zostać przeprowadzona bez interakcji pmiędzy nimi.

W przypadku aplikacji ransomware botmaster ma do czynienia tylko z jednym problemem związanym z komunikacją: z generowaniem i dystrybucją kluczy.

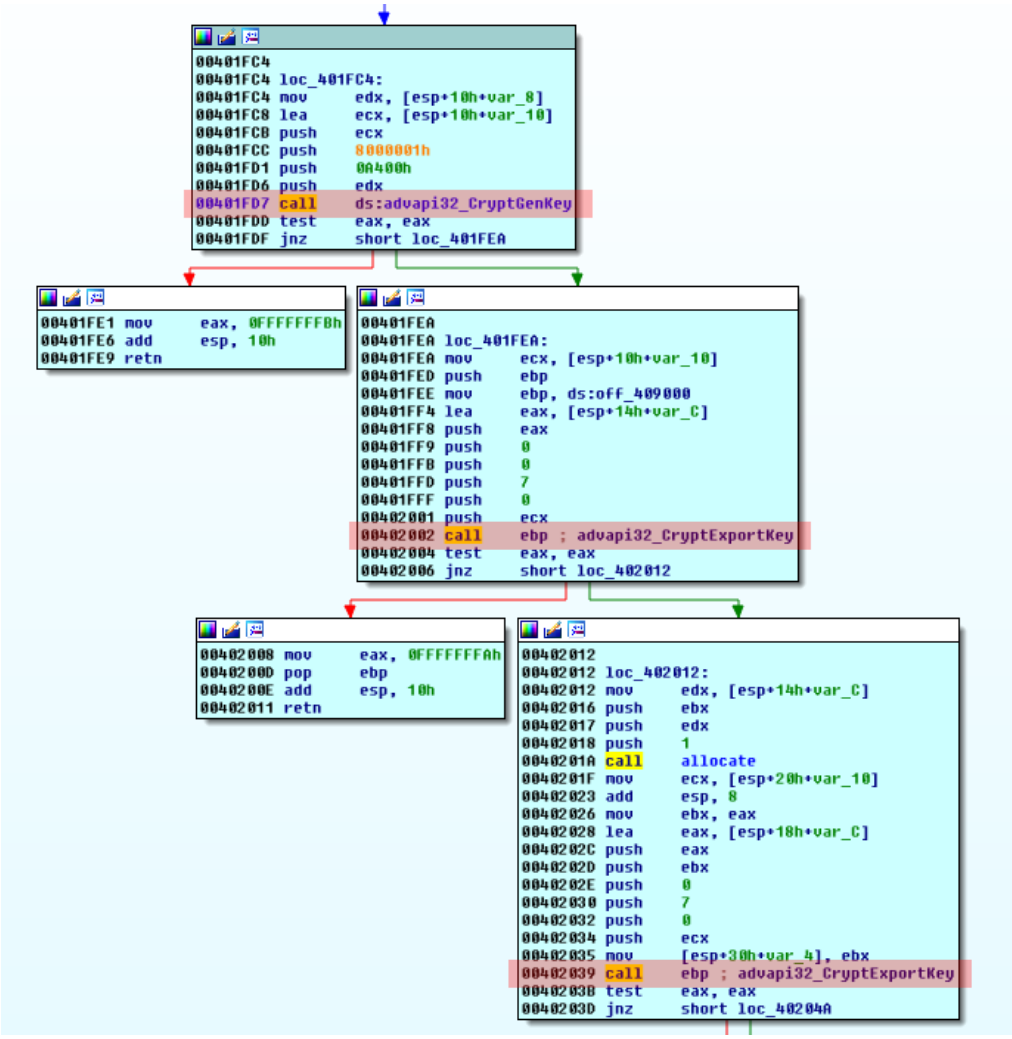
Po pierwsze, w przypadku ransomware kryptografia symetryczna nie nadaje się do zastosowania. Wynika to z tego, że po pozyskaniu próbki w laboratorium podczas analizy procesu szyfrowania można uzyskać klucz i algorytm potrzebny do odszyfrowania danych, a co za tym idzie – do porażki kampanii. Z tego powodu twórcy oprogramowania ransomware opierają się na kryptografii asymetrycznej.

Drugą kwestią, którą należy wziąć pod uwagę w związku z kryptografią jest fakt, że każda para kluczy powinna być wygenerowana indywidualnie dla konkretnej ofiary. Jeśli wszystkie ofiary zostałyby zaatakowane z wykorzystaniem tej samej pary kluczy, mogłoby dojść do sytuacji, gdy jedna z ofiar po wpłacie okupu udostępnia klucz deszyfrujący pozostałym ofiarom. A brak konieczności zapłaty okupu oznacza porażkę cyberprzestepcy.

Z tych powodów aplikacje ransomware generują indywidualne pary kluczy kryptografii asymetrycznej. W części próbek proces ten był wykonywany za pośrednictwem bezpośredniej komunikacji bota z botmasterem.

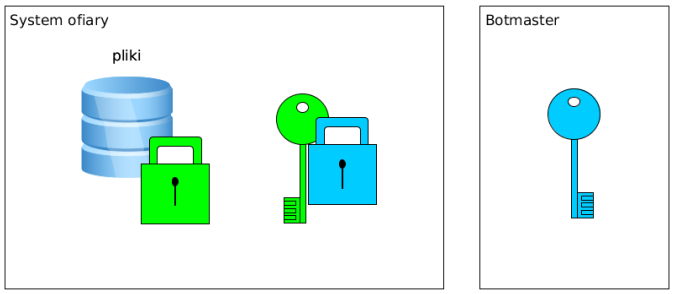
Przykład:

Bot po infekcji systemu ofiary odnajdywał unikalny ciąg bajtów pozwalający na jednoznaczną identyfikację systemu, np. GUID, przekazując go następnie botmasterowi za pośrednictwem kanału C&C. Botmaster generował unikalną parę kluczy i przypisywał do nich unikalny identyfikator, część publiczną przekazując za pośrednictwem kanału C&C do bota, który używał jej do szyfrowania danych. Po zaszyfrowaniu danych, poinformowaniu ofiary i wpłaceniu okupu, botmaster na podstawie podanego przez ofiarę identyfikatora udostępniał prywatną część klucza, pozwalającą ofierze na odszyfrowanie danych. Autorzy Cry wprowadzili nowy schemat, który umożliwił im zminimalizowanie interakcji z systemem ofiary, jednocześnie nie rezygnując z generowania indywidualnych par kluczy asymetrycznych.



Najpierw bot generuje indywidualną parę kluczy asymetrycznych (na potrzeby tego raportu nazywaną parą operacyjną). Część publiczna jest używana do szyfrowania danych w systemie ofiary, prywatna zaś jest szyfrowana zakodowaną w próbce częścią publiczną drugiej pary kluczy (zwanej parą główną). Zaszyfrowany klucz jest dodatkowo przekształcany w ciąg znaków, który stanowi identyfikator ofiary.

By lepiej zrozumieć stosowany schemat wyobraźmy sobie części publiczne kluczy jako kłódki, zaś części prywatne jako klucze do nich. Załóżmy, że pary operacyjne są koloru zielonego, a pary główne – niebieskiego. Posługując się tą reprezentacją, można powiedzieć, że autorzy Cry za pomocą swojego programu dostarczają do systemu ofiary niebieską kłódkę. Po uruchomieniu programu, generuje on indywidualną zieloną kłódkę i zielony klucz. Zielona kłódka jest używana do zamknięcia danych ofiary, a niebieska kłódka jest używana do zamknięcia zielonego klucza.



Zielona kłódka jest inna dla każdej ofiary, tak więc po uzyskaniu swojego zielonego klucza, jedna ofiara nie może przekazać go do wykorzystania innej. Aby uzyskać dostęp do swojego zielonego klucza, ofiara musi wysłać zamknięty niebieską kłódką zielony klucz do botmastera przy wpłacaniu okupu. Używając swojego niebieskiego klucza botmaster otwiera i odsyła ofierze zielony klucz.

Kanał C&C

Pomimo tego, że wymagania komunikacji pomiędzy botem i botmasterem są ograniczone do minimum, oprogramowanie to posiada jednak dość rozbudowany kanał C&C, składający się z dwóch podkanałów.

Podkanał 1

Opiera się na protokole UDP. W trakcie wykonywania operacji, próbka formułuje raporty, które wysyła za pomocą tego podkanału do ponad 1000 systemów o numerach IP wygenerowanych za pomocą wbudowanego algorytmu będącego odpowiednikiem DGA (Domain Generation Algorithm). Dzięki takiemu rozwiązaniu, zainfekowane systemy ofiar generują dużą ilość ruchu maskującego, który utrudnia zlokalizowanie odbiorcy kanału C&C. Ponieważ protokół UDP jest bezpołączeniowy, można jedynie spekulować, że co najmniej jeden z wygenerowanych adresów IP należy do infrastruktury botmastera. Aby to jednak zweryfikować, należałoby dotrzeć i przeanalizować każdy z nich.

Niewykluczone również, że żaden z wygenerowanych adresów nie należy do botmastera, a generowany ruch ma na celu jedynie zmylenie służb zajmujących się ściganiem tego typu przestępstw.

Podkanał 2

Podkanał 2 to serwis w sieci TOR, w praktyce jedyny niezbędny kanał komunikacyjny. Służy on do przekazania okupu i identyfikatora ofiary botmasterowi i w odpowiedzi do uzyskania od niego klucza deszyfrującego. Dokładny opis, jak korzystać z tego kanału, znajduje się w notatce umieszczanej na pulpicie ofiary.

Można jedynie spekulować na temat przyczyn umieszczenia w próbce aż dwóch podkanałów C&C. Dokładne przeanalizowanie zakończeń podkanału 1 wymagałoby akwizycji dowodów z ponad 1000 systemów, a nawet to nie gwarantuje, że umożliwiłyby one zlokalizowanie infrastruktury botmastera. Prawdopodobną możliwością jest zbieranie danych na temat ofiar, które nie zdecydowały się na kontakt w celu uiszczenia okupu lub złośliwe oprogramowanie odstąpiło od procesu infekcji. Więcej informacji o możliwości odstąpienia od infekcji znajduje się w dalszej części raportu.

Inne uwagi

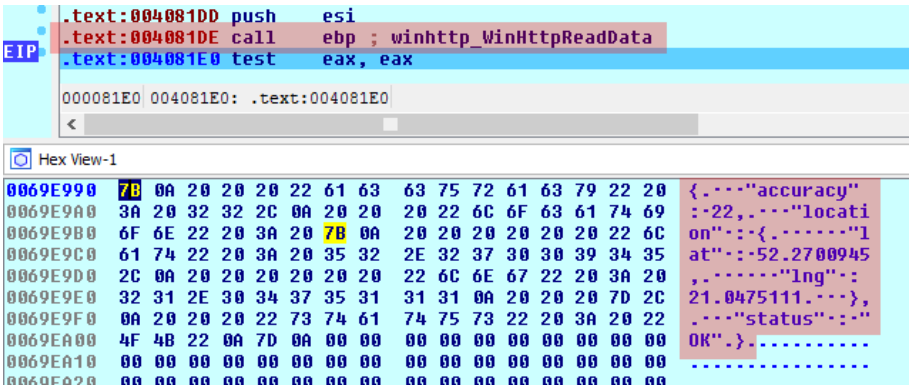
Cry posiada kilka elementów, które odróżniają go od innych programów tej klasy.

Generowanie nazw

Podczas działania Cry tworzy szereg wspierających jego operacje obiektów. Mogą one zostać wykorzystane do wykrycia infekcji (ang. IoC – Indicators of Compromise). Aby tego uniknąć, w próbce zaimplementowano funkcję generowania pseudolosowych ciągów znaków, wykorzystywanych następnie do tworzenia obiektów. Jako dane wejściowe generatora wprowadzany jest wygenerowany unikalny GUID infekowanego systemu, co zapewnia, że dla każdej infekcji nazwy będą się różniły.

Analiza WLAN

Próbka posiada funkcję badania, czy system ofiary jest podłączony do sieci bezprzewodowych, by pograć informację o znajdujących się w zasięgu sieciach radiowych, a następnie spróbować połączyć się z API Google Maps w celu geolokalizacji systemu ofiary.



Po uzyskaniu tej informacji, wysyła ją w formie raportu do podkanału 1 C&C.

Odstąpienie od infekcji

Cry posiada wbudowany mechanizm zabezpieczający, który przewiduje odstąpienie od infekcji ofiary w przypadku spełnienia pewnych warunków. {rzed podjęciem decyzji o infekcji pobierane są dane na temat używanych języków klawiatury.

```
.text:00402EA9 push    0Ah
.text:00402EAB call    ds:user32_GetKeyboardLayoutList
.text:00402EB1 test    eax, eax
.text:00402EB3 jle     short loc_402EF8
.text:00402EB5 xor     edx, edx
.text:00402EB7 test    eax, eax
.text:00402EB9 jle     short loc_402EF8
.text:00402EBB jmp     short loc_402EC0
.text:00402EBD ; ---------------------------------------
.text:00402EBD lea     ecx, [ecx+0]
.text:00402EC0
.text:00402EC0 loc_402EC0:
.text:00402EC0
.text:00402EC0 mov     cx, [esp+edx*4+2Ch+var_28]
.text:00402EC5 mov     esi, 3FFh
.text:00402ECA and     cx, si
.text:00402ECD movzx   ecx, cx
.text:00402ED0 cmp     ecx, 23h
.text:00402ED3 jz      short loc_402EFF
.text:00402ED5 cmp     ecx, 3Fh
.text:00402ED8 jz      short loc_402EFF
.text:00402EDA cmp     ecx, 19h
.text:00402EDD jz      short loc_402EFF
.text:00402EDF cmp     ecx, 22h
```

Jeśli system wykorzystuje język białoruski, kazachski, rosyjski, ukraiński, uzbecki lub estoński, program decyduje się jedynie na wysłanie raportu o podwyższeniu uprawnień w systemie do podkanału 1 C&C, ale odstępuje od szyfrowania danych na dyskach i usuwa sam siebie.

Protektor

Właściwy kod aplikacji Cry jest zabezpieczony przed analizą. Zanim nastąpi jego wykonanie, konieczne jest jego odbezpieczenie. Protektor wykorzystywany w przypadku niniejszej próbki nie odbiega znacząco od innych próbek złośliwego oprogramowania. W pierwszym kroku importowane są wywołania biblioteczne umożliwiające alokację pamięci i rozpakowanie właściwego kodu. Do zaalokowanych fragmentów przenoszony jest, a następnie wykonywany przetwarzany kod.

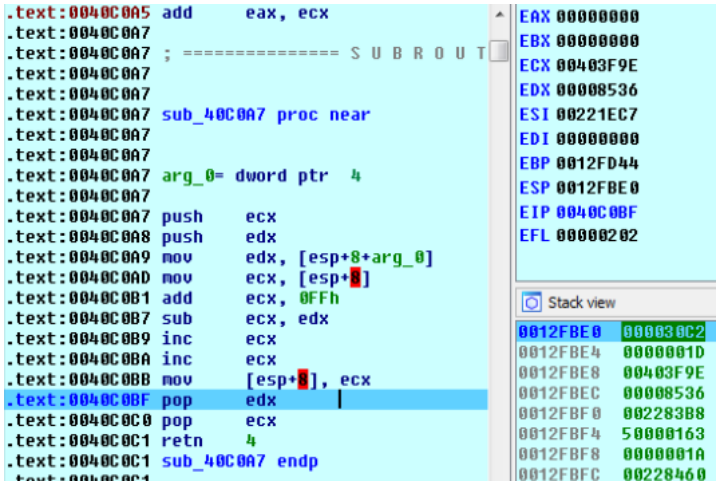
Jedną z ciekawych konstrukcji jest funkcja modyfikująca własny adres powrotu na podstawie zadanych argumentów.

```
.text:0040C0A5 add     eax, ecx
.text:0040C0A7
.text:0040C0A7 ; =============== S U B R O U T
.text:0040C0A7
.text:0040C0A7
.text:0040C0A7 sub_40C0A7 proc near
.text:0040C0A7
.text:0040C0A7
.text:0040C0A7 arg_0= dword ptr  4
.text:0040C0A7
.text:0040C0A7 push    ecx
.text:0040C0A8 push    edx
.text:0040C0A9 mov     edx, [esp+8+arg_0]
.text:0040C0AD mov     ecx, [esp+8]
.text:0040C0B1 add     ecx, 0FFh
.text:0040C0B7 sub     ecx, edx
.text:0040C0B9 inc     ecx
.text:0040C0BA inc     ecx
.text:0040C0BB mov     [esp+8], ecx
.text:0040C0BF pop     edx
.text:0040C0C0 pop     ecx
.text:0040C0C1 retn    4
.text:0040C0C1 sub_40C0A7 endp
.text:0040C0C1
```

```
EAX 00000000
EBX 00000000
ECX 00403F9E
EDX 00008536
ESI 00221EC7
EDI 00000000
EBP 0012FD44
ESP 0012FBE0
EIP 0040C0BB
EFL 00000202

Stack view
0012FBE0  000030C2
0012FBE4  0000001D
0012FBE8  0040C3D3
0012FBEC  00008536
0012FBF0  002283B8
0012FBF4  50000163
0012FBF8  0000001A
0012FBFC  00228460
```

Po wywołaniu tej funkcji z określonymi argumentami pobiera ona ze stosu adres powrotu i wykonuje na nim działania arytmetyczne, a wynikiem nadpisuje adres powrotu.



W ten sposób funkcja, zamiast autonomicznej jednostki wykonywanych operacji, spełnia rolę skomplikowanej instrukcji skoku do innej części kodu. Takie zastosowanie funkcji może posłużyć do zmylenia narzędzi, które zakładają tradycyjne wykorzystanie instrukcji call.

Odzyskiwanie plików

Ze względu na błędy popełniane w procesie szyfrowania, pliki zaszyfrowane z wykorzystaniem analizowanej próbki są możliwe do odzyskania w drodze analizy z zakresy informatyki śledczej. Użytkownik, który padł ofiarą aplikacji Cry może sprobować odzyskać te dane samodzielnie (na przykład korzystając z darmowego pakietu Sleuthkit przeznaczonego dla systemu Linux) lub skorzystać z usług jednej z kilku firm zajmujących się odzyskiwaniem danych.